EDA

TPDATA Interface User's Guide

Contents

TPDATA Interface

The TP family of interfaces enables the retrieval and maintenance of mainframe data. Currently, it includes two types of interfaces, TPVSAM and TPIMSDB.

The TPDATA Interfaces execute from Information Builders' Cactus/Maintain or FOCUS 4 GL languages via the EDA, Cactus, or FOCUS product lines. The FOCUS product line consists of FOCUS on all platforms including the FOCUS Desktop product and Mainframe FOCUS. The TPDATA Interfaces are designed to run from any client platform retrieving mainframe data with little or negligible mainframe resource utilization. This negligible mainframe resource utilization sets these Interfaces apart from other mainframe data retrieval and update products.

The TPDATA Interfaces enable CICS to manage the transactions. This reduces the amount of memory required per user and significantly increases the maximum number of concurrent users.

Note: Two additional interfaces are planned one to access IMS data that is controlled by IBM's IMS/TM region which will be called TPIMSTM and TPDB2 to access DB2 via CICS.

Your entire application can run in a UNIX, Windows NT, MVS, CMS, or AS400 environment while transactions are executed on the mainframe. In this document, the mainframe refers to any platform where your CICS region resides such as MVS, OS390, or VSE/ESA. Your client can be a Web browser, Cactus, Mainframe FOCUS, FOCUS Desktop, Mainframe Maintain, or any EDA/Client. The server can be Mainframe FOCUS or any EDA Server on any platform. The TPDATA Interfaces to IMS and VSAM enable applications to read and update operational data and gain the scalability of the UNIX or Window NT environment.

Note: The EDA/TP Gateway for CICS must be installed before you can use the TPDATA Interfaces for VSAM, IMS/CICS and DB2.
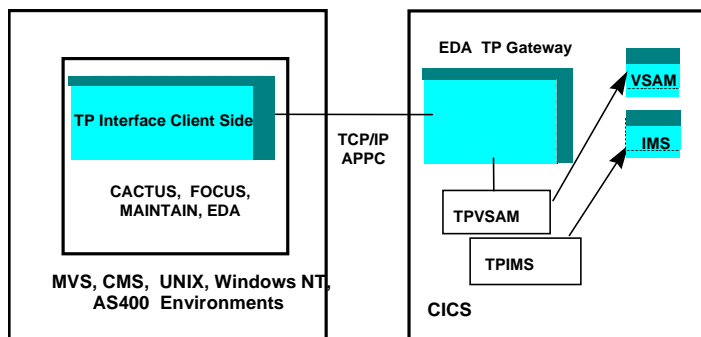
TPDATA Interface Components

The TPDATA Interface consists of three distinct components:

- **TPDATA Interface server software components**. The server programs reside and run within the CICS region. The program installed depends on the Interface licensed: program TPVSAM for the TPVSAM Interface or program TPIMSDB for the CICS/IMS Interface. Included on the EDA/TP Gateway tape are the object modules for these programs, the JCL to link the object modules, and all of the required CICS CSD entries and JCL to enable the Interface to execute within the CICS region.

- **TPDATA Interface client software**. The TPDATA Interface client program, VVCICSET, the required client program, resides in the EDA Server or Mainframe FOCUS environment.

- **TPDATA Interface sample data files**. The tape includes sample Master Files and Access Files for two well-known VSAM and IMS data sources. The VSAM file called DFHFILEA and the IMS file called DI21PART are defined. These sample Master Files and Access Files can be ported by FTP, or another file transfer product, to the EDA Server platform—for example, Windows NT or UNIX. A Cactus Web application running on NT and accessing VSAM or IMS data requires that the Master and Access Files are physically located on the Cactus NT EDA Server.

  Use the supplied Master and Access Files to verify the installation of the TPDATA Interfaces. For the TPVSAM Interface, the native VSAM file, FILEA, must be allocated correctly to the CICS region. Similarly, for the TPIMSDB Interface, the native IMS file DI21PART must be allocated correctly to the IMS DBCTL region associated with the CICS region.

Reference   TPDATA Interface Server Components

The server installation components included on the EDA/TP Gateway installation tape are:

	TPVSAM Interface	**TPIMSDB Interface**
TPDATA Interface Server program object modules	TPVSMOBL TPVSMOBP	TPIMSOBL TPIMSOBP
TPDATA Interface Server program MVS JCL members to link the above programs	TPVSMLNK	TPIMSLNK
TPDATA Interface Server CICS CSD update members	TPVSMENT	TPIMSENT
TPDATA Interface CICS CSD update MVS JCL members	TPVSMCSD	TPIMSCSD

Reference   TPDATA Interface Sample Master and Access Files

The following sample Master and Access Files are delivered on the MVS EDA/TP Gateway tape:

Master and Access Files for …	**Master File Member**	**Access File**
CICS/VSAM file FILEA	TPFILMAS	TPFILACX
CICS/VSAM file FILEA	TPFILMAS	TPFILFCX*
CICS/IMSDB file DI21PART	TPD21MAS	TPD21ACX
CICS/IMSDB file DI21PART	TPD21MAS	TPD21FCX*

* Use this Master File when communicating with a Windows NT, UNIX, or MVS EDA Server. For details see ***Configuring Communications With an EDA Server***.

Overview of the TPDATA Interface Installation Process

Before you can use the TPDATA Interface you must complete the following process:

- Verify that the EDA/TP Gateway for CICS is fully operational. See ***TPDATA Interfaces and the EDA/TP Gateway***.
- Install the server components for MVS. See ***Installing the TPDATA Interface Server Components***.
- Install the client component of the TPDATA Interface. See ***Installing the TPDATA Interface Client Component***.
- Define your Master Files and Access Files. See ***Defining Master Files and Access Files***.
- If you are communicating with an EDA Server, see ***Configuring Communications With an EDA Server*** for additional steps.

TPDATA Interfaces and the EDA/TP Gateway

The TP family of interfaces is designed to take advantage of the mainframe multi-user address spaces such as the IBM's CICS and the IMS/TM (IMS/DC) environments.

The TPDATA Interfaces use the EDA/TP Gateway as the connection to the CICS region. The EDA/TP Gateway product supports:

- CICS/MVS or OS390 version 4.1 and above.
- CICS/VSE 2.3 and above.

The remainder of this topic describes how to install the TPVSAM and TPIMSDB Interfaces for VSAM or IMS data controlled by an IBM MVS CICS region. The EDA/TP Gateway must be completely operational before installing any TPDATA Interface.

Refer to the *EDA/TP Gateway for CICS* manual for information about specific software compatibility issues and to ensure that you have fully verified the EDA/TP gateway installation and configuration before you install TPDATA Interfaces.

Installing the TPDATA Interface Server Components

The TPDATA Interface server components were unloaded as part of the EDA/TP Gateway for CICS installation procedure. The JCL members and input sources reside in the *qualif*.ETPGATE.DATA library. The object modules for the TPVSAM and TPIMSDB programs reside in the *qualif*.ETPGATE.LOAD library.

The installation process for server components includes three steps:

1. Add group TPDATA to CICS startup list. See ***How to Add Group TPDATA to the Startup List***.
2. Configure the server component for your data source. (Configuration includes a linking step followed by an updating step.)

   To configure for VSAM, see ***Configuring for CICS/VSAM***.

   To configure for IMS, see ***Configuring for CICS/IMS***.
3. Modify the CICS region startup JCL. See ***Modifying the CICS Region Startup JCL***.

Syntax How to Add Group TPDATA to the Startup List

To include the TPDATA group in the list of groups to activate when the CICS region starts, edit the member ETPGATE.DATA(TPDATCSD) and submit the job. The JCL is

```
//*----------------------------------------------------------------*
//* Add TPDATA group to CICS region                                *
//*----------------------------------------------------------------*
//DFHCSDUP EXEC PGM=DFHCSDUP,REGION=2048K,
//         PARM='CSD(READWRITE),PAGESIZE(60),NOCOMPAT'
//STEPLIB  DD DSN=qualif.SDFHLOAD,DISP=SHR
//*point to your region
//DFHCSD   DD DSN=qualif.DFHCSD,DISP=SHR
//*
//SYSPRINT DD SYSOUT=*
//SYSIN    DD  *
ADD GROUP(TPDATA) LIST(STARTUPLIST)
/*
//
```

where:

qualif

Is the high-level qualifier for the CICS CSD to be updated.

Configuring for CICS/VSAM

You must link the TPDATA VSAM access program object modules TPVSMOBL and TPVSMOBP to create the program load module TPVSAM. The JCL member *qualif*.ETPGATE.DATA (TPVSMLNK) creates the server program.

Syntax How to Link the TPDATA VSAM Access Program for CICS

To link the program TPVSAM module, edit and run the member TPVSMLNK in *qualif*.ETPGATE.DATA.The
sample JCL is

```
//* job card goes here
//*----------------------------------------------------------------*
//*    LINK THE TPDATA VSAM ACCESS PROGRAM                         *
//*----------------------------------------------------------------*
//LINKVSM   PROC
//*
//LKED   EXEC  PGM=IEWL,PARM='LIST,XREF,LET,MAP',
//*  ALLOCATE THE FOLLOWING LIBRARY TO RPL OF THE CICS REGION
//SYSLMOD  DD  DSNAME=prefix.ETPGATE.LOAD(&PROGNAME),DISP=SHR
//*
//SYSLIB   DD  DSNAME=SYS1.COB2CICS,DISP=SHR
//         DD  DSN=qualif.SDFHLOAD,DISP=SHR
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=*
//OBJECT   DD  DSN=prefix.ETPGATE.LOAD,DISP=SHR
//SYSLIN   DD  DDNAME=SYSIN
//         PEND
//*----------------------------------------------------------------*
//*    LINK THE TPDATA VSAM ACCESS PROGRAM                         *
//*----------------------------------------------------------------*
//CPGCICS  EXEC LINKVSM,PROGNAME='TPVSAM'
//LKED.SYSIN DD *

INCLUDE SYSLIB(DFHECI)
INCLUDE OBJECT(TPVSMOBL)
INCLUDE OBJECT(TPVSMOBP)
MODE AMODE(31),RMODE(ANY)
NAME TPVSAM(R)
/*
//
```

where:

prefix

> Is the prefix for the TP Gateway libraries.

SYSLMOD

> Points to the library where the program TPVSAM is to be placed. This is allocated to ddname RPL in the CICS
> region.

Syntax How to Update the CICS System Dataset (CSD) for VSAM

The JCL member *qualif*.ETPGATE.DATA(TPVSMCSD) updates the CSD with the definitions required for the TP
data access program. To update the CSD to define the TPDATA Interface server program TPVSAM, edit the member
and submit the job. The JCL is:

```
//*----------------------------------------------------------------*
//* UPDATE CSD FOR TPDATA VSAM ACCESS PROGRAM                      *
//*----------------------------------------------------------------*
//DFHCSDUP EXEC PGM=DFHCSDUP,REGION=2048K,
//         PARM='CSD(READWRITE),PAGESIZE(60),NOCOMPAT'
//STEPLIB  DD DSN=qualif.SDFHLOAD,DISP=SHR
//DFHCSD   DD DSN=qualif.CICS410.DFHCSD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DISP=SHR,DSN=prefix.ETPGATE.DATA(TPVSMENT)
```

where:

qualif

> Is the high-level qualifier for the CICS CSD to be updated.

prefix

Is the prefix for the TP Gateway libraries.

Input to JCL job ETPGATE.DATA(TPVSMCSD)

```
00001 DEFINE PROGRAM(TPVSAM) GROUP(TPDATA)
00002 DESCRIPTION(COBOL VSM LOGICAL AND PHSICAL SERVERS)
00003       LANGUAGE(COBOL) RELOAD(NO) RESIDENT(NO) USAGE(NORMAL)
00004       USELPACOPY(NO) STATUS(ENABLED) CEDF(YES) DATALOCATION(ANY)
00005       EXECKEY(USER) EXECUTIONSET(FULLAPI)
```

Configuring for CICS/IMS

You must link the TPDATA IMS access program object modules TPIMSOBL and TPIMSOBP to create the program load module TPIMSDB. The JCL member *qualif*.ETPGATE.DATA (TPIMSLNK) creates the server program.

Syntax   How to Link the TPDATA IMS Access Program for CICS

To link the program TPIMSDB module, edit and run the member TPIMSLNK in *qualif*.ETPGATE.DATA. The sample JCL is

```
//* job card goes here
//*----------------------------------------------------------------*
//*    LINK THE TPDATA IMS ACCESS PROGRAM                          *
//*----------------------------------------------------------------*
//LINKIMS   PROC
//*
//LKED   EXEC  PGM=IEWL,PARM='LIST,XREF,LET,MAP',
//*  ALLOCATE THE FOLLOWING LIBRARY TO RPL OF THE CICS REGION
//SYSLMOD  DD  DSNAME=prefix.ETPGATE.LOAD(&PROGNAME),DISP=SHR
//*
//SYSLIB   DD  DSNAME=SYS1.COB2CICS,DISP=SHR
//         DD  DSN=qualif.SDFHLOAD,DISP=SHR
//SYSUT1   DD  UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD  SYSOUT=*
//OBJECT   DD  DSN=prefix.ETPGATE.LOAD,DISP=SHR
//SYSLIN   DD  DDNAME=SYSIN
//         PEND
//*----------------------------------------------------------------*
//*    LINK THE TPDATA IMS ACCESS PROGRAM                          *
//*----------------------------------------------------------------*
//CPGCICS  EXEC LINKIMS,PROGNAME='TPIMSDB'
//LKED.SYSIN DD *
INCLUDE SYSLIB(DFHECI)
INCLUDE OBJECT(TPIMSOBL)
INCLUDE OBJECT(TPIMSOBP)
MODE AMODE(31),RMODE(ANY)
NAME TPIMSDB(R)
/*
//
```

where:

prefix

Is the prefix for the TP Gateway libraries.

SYSLMOD

Points to the library where the program TPIMSDB is to be placed. This is allocated to ddname RPL in the CICS region.

Syntax How to Update the CICS System Dataset (CSD) for IMS

The JCL member *qualif*.ETPGATE.DATA(TPIMSCSD) updates the CSD with the definitions required for the TP data access program. To update the CSD to define the TPDATA Interface server program TPIMSDB, edit the member and submit the job. The JCL is

```
//*------------------------------------------------------------------*
//* UPDATE CSD FOR TPMSDB ACCESS PROGRAM                             *
//*------------------------------------------------------------------*
//DFHCSDUP EXEC PGM=DFHCSDUP,REGION=2048K,
//          PARM='CSD(READWRITE),PAGESIZE(60),NOCOMPAT'
//STEPLIB  DD DSN=qualif.SDFHLOAD,DISP=SHR
//DFHCSD   DD DSN=qualif.CICS.DFHCSD,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DISP=SHR,DSN=prefix.ETPGATE.DATA(TP)
```

where:

qualif

> Is the high-level qualifier for the CICS CSD to be updated.

prefix

> Is the prefix for the TP Gateway libraries.

Input to JCL job ETPGATE.DATA(TPIMSCSD)

```
00001 DEFINE PROGRAM(TPIMSDB) GROUP(TPDATA)
00002 DESCRIPTION(COBOL VSM LOGICAL AND PHSICAL SERVERS)
00003       LANGUAGE(COBOL) RELOAD(NO) RESIDENT(NO) USAGE(NORMAL)
00004       USELPACOPY(NO) STATUS(ENABLED) CEDF(YES) DATALOCATION(ANY)
00005       EXECKEY(USER) EXECUTIONSET(FULLAPI)
```

Modifying the CICS Region Startup JCL

Once you have added group TPDATA to the JCL startup list and configured the server component for your data source, you are ready to modify the CICS startup JCL that contains the TP Gateway to concatenate the TPDATA Interface load library. This load library contains the TPVSAM or TPIMSDB modules previously created in the CICS region startup JCL.

Syntax   How to Modify the CICS Region Startup JCL

Modify the CICS startup JCL as follows:

```
//*------------------------------------------------------------------*
//* qualif.CICS.JCLLIB(REGION41)      *
//*------------------------------------------------------------------*
//DFHRPL   DD DSN=prefix.ETPGATE.LOAD,DISP=SHR
//         DD DSN=WIBFOC.R72XFOC.EDATPG.LOAD,DISP=SHR
//         DD DSN=&INDEX2..SDFLOAD,DISP=SHR
//         DD DSN=qualif.CICS.LOADLIB,DISP=SHR
//         DD DSN=SYS1.COB2CICS,DISP=SHR
//         DD DSN=SYS1.COB2LIB,DISP=SHR
//         DD DISP=SHR,DSN=TCPIP.V3R1.SEZATCP
//         DD DSN=INTRLINK.V200CPT.T09LOAD,DISP=SHR
```

where:

qualif

> Is the high-level qualifier for the CICS region.

prefix

> Is the prefix for the TP Gateway libraries.

Installing the TPDATA Interface Client Component

The TPDATA Interface client component for the TPVSAM and TPIMSDB Interface consists of one module called VVCICSET. This module resides and is executed on the EDA Server or Mainframe FOCUS environment. If the front-end application, such as a Web browser, communicates with an EDA Server on UNIX, then the Master and Access Files reside on that UNIX box.

There is a VVCICSET module for Windows NT, UNIX, and MVS. A Cactus Web application or a FOCUS Desktop application, running on NT and accessing VSAM or IMS data, requires that the DLL VVCICSET be located in the appropriate path on the Cactus NT Server. For a Mainframe FOCUS application, the load module VVCICSET must be allocated to USERLIB, FOCLIB, or STEPLIB of the TSO logon procedure or MVS batch job. When an EDA Server is used for communications, the VVCICSET module is installed during the EDA Server installation process.

Reference   Valid Client Environments for TPDATA

The following table includes minimum IBI software release levels.

Client	Information Builders Software	Installation Directory
FOCUS Desktop (for Windows NT and MS-95)	FOCUS Desktop 4.x	Module VVCICSET is downloaded as part of the FOCUS Desktop installation process.
Windows NT and UNIX (Cactus or EDA client)	Cactus or EDA Server 4.x	Module VVCICSET is downloaded as part of the Cactus and EDA Server 4.x installation process.
Mainframe FOCUS client (MVS CLIST or BATCH)	Mainframe FOCUS 7.09	Module VVCICSET resides on the FOCUS for Mainframe Installation tape. Refer to *Mainframe FOCUS New Feature 693* for instructions on how to download the appropriate TPDATA Client Interface program.
EDA Server for MVS client	EDA Server 4.x	Module VVCICSET is downloaded as part of the EDA Server 4.x installation process. It resides in library qualif.EDALIB.LOAD.

Note: This software also runs with WebFOCUS, using an EDA Server. The location of the EDA Server dictates the appropriate installation directory.

Defining Master Files and Access Files

IBI products driven by a Master File and an Access File are capable of accessing and maintaining mainframe data via a TPDATA Interface. Master and Access Files reside on the server with a special SUFFIX value. This suffix is used to execute the appropriate TPDATA Interface. When a Master File contains SUFFIX=TPVSAM or TPIMS, the directories or libraries are searched for in the appropriate TPDATA Client Interface. For SUFFIX values of TPIMS and TPVSAM, DLL or load module CICSVVSET is executed.

These Master Files and Access Files can reside on Windows NT, UNIX, or MVS server platforms and are identified as follows:

FOCUS or EDA Platform	Master Files	Access Files
Windows NT and UNIX	Have a .MAS extension.	Have an .ACX extension.
Mainframe FOCUS	Allocated to ddname MASTER	Allocated to ddname ACCESS
MVS EDA Server 4.x	Allocated to ddname EDAMFD	Allocated to ddname EDAAFD

The supplied Master and Access Files must have the same name with either a .MAS or .ACX extension. For Mainframe FOCUS or an EDA MVS Server, the Master and Access Files must also have the same name and reside in their correct libraries.

For more specific information about where these Master and Access Files must reside, see the FOCUS or EDA documentation for your operating environment.

TPDATA Master Files

The TPDATA Master File is a description of the native VSAM or IMS file to be accessed by the TPDATA Interface. The TPDATA Client Interface uses the Master File as well as its corresponding Access File to access and maintain the corresponding data type. The Master File describes the field names and lengths of all the fields of the underlying data

record. The Master File also contains a value for the SUFFIX attribute that specifies what interface its going to execute. A SUFFIX value of TPVSAM denotes the TPVSAM Interface; a SUFFIX value of TPIMS the TPIMSDB Interface.

You must describe all data sources (VSAM, IMS and DB2)—including data sources with multiple record types—as you would a flat VSAM Master File. Because all data sources must be represented as a VSAM-like structure, Master Files for the TPVSAM Interface are identical to other VSAM Master Files that may already exist at your site. The only difference is the addition of the CODEPAGE parameter. The CODEPAGE value must be 37. The CODEPAGE value of 37 is required when this Master File is on a platform other than MVS to allow for data conversion.

There are other considerations for IMS and DB2, which are explained below for each of the data types.

For more information about describing multiple record types in a Master File, see related topics in your FOCUS or Cactus documentation.

Note: A tool to automate the creation of Master Files and Access Files is planned but not yet developed.

Example Sample TPDATA VSAM Master File

The following is a sample Master File for the CICS/VSAM file FILEA.

The following Master File member TPFILMAS was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

The supplied Master File for the VSAM file FILEA is identical to other VSAM Master Files that may already exist at your site with one difference. Suffix=TPVSAM executes the TPVSAM Interface.

Keys and indices of a VSAM file are indicated in the master file with ALIAS=KEY for keys and INDIEX=I for the alternate index. When an IF or WHERE selection test on one of these fields is present in a request, the TPDATA client will pass the IF/WHERE condition to the server. The limited answer set will limit the size of the buffer passed which in turn limits the total size of the CICS transaction. The use of IF/WHERE tests is highly recommended to limit resource utilization and the impact on the CICS region.

```
FILE=FILEA,SUFFIX=TPVSAM,CODEPAGE=37,$
SEGNAME=FILEA,SEGTYPE=S0,$
 FIELD=FILL, ,A1,  A1,$
 GROUP=IDKEY,ALIAS=KEY,USAGE=A6,  ACTUAL=A6,$
 FIELD=ID1, ID1A,A3,  A3,$
 FIELD=ID2, ID2A,A3,  A3,$
 FIELD=NAME, NAME, A20, A20,$
```

Example Sample TPDATA IMS Master File

The following is a sample Master File for the CICS/IMS file DI21PART.

The following Master File member TPD21MAS was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

Please note the indented lines denote special segment prefixes that are required for the TPDATA Interface for IMS. These fields need to be added manually for already created IMS Master Files or Master Files that are created by the Cobol FD Translator. The prefix represents the IMS key feedback area which prefixes the segment data returned by IMS. Each segment in a TPIMS Master File must begin with fields whose total length equals the maximum key feedback length returned by IMS. For this example, the key feedback area length is 44 bytes. The length can be obtained from the PSB source. A group field denoted by GROUP= is the start of the IMS segment prefix. For instance, the field GROUP1 describes the length of the concatenated key returned for that IMS segment. It also must contain an ALIAS=KEY. In the following TPIMS Master File for the IMS file DI21PART, all segment prefixes are denoted with the name GROUP1,2,3, etc. The length of the group field is the total length of it's parent key plus the length of the segment key. For child segment STANINFO, the length of the parent key plus the length of the IMS key length of segment STANINFO is 17 + 2 = 19. Field PLACE_HOLDER is used to account for the length of the parent segment's key. The FILLER fields such as FILEER1,2,3 etc account for the remainder of the maximum key feedback length. For example, segment STANINFO has a true key length of 2 bytes for the key field, STAN_KEY, the total of lengths of fields PLACE_HOLDER + STAN_KEY + FILLER2 must equal 44, the maximum key feedback area length.

The lines indicated in pink denote the actual IMS segment name. This segment is described to the Interface by using the FIELD=RECTYPE construct. The usage and actual of the FIELD=RECTYPE is always equal to 8 which is the maximum IMS segment name length. The fields following the FIELD=RECTYPE is the data portion of the IMS segment which describes the IMS segment which can be generated by the Cobol FD Translator or manually created out of a BAL DSECT. The root segment must contain, within the data portion, a field that contains an ALIAS=ims-key-name,INDEX=I so that the Interface may build an SSA to qualify data retrieval.

In this case since PARTKEY is defined to IMS as the key to the root of the segment, FIELD=PARTKEY,ALIAS=PARTKEY, INDEX=I is required.

Keys and indices of an IMS file are indicated in the Master File with INDEX=I for both IMS keys and any alternate indices. When an IF or WHERE selection test on one of these fields is present in a request, the TPDATA client will pass the IF/WHERE condition to the server. The limited answer set will limit the size of the buffer passed which in turn limits the total size of the CICS transaction. The use of IF/WHERE tests is highly recommended to limit resource utilization and the impact on the CICS region.

A request that contains an IF or WHERE test on field PART_KEY (denoted with INDEX=I) the Interface will create a qualified SSA to pass to IMS. Also, ranges or retrieval requests without selection criterion on an IMS HDAM file will return an error.

```
FILE=DI21PART,SUFFIX=TPIMS,CODEPAGE=37,$
 SEGMENT=PARTROOT,SEGTYPE=S0,$
  GROUP =GROUP1,ALIAS=KEY,A17,A17,$
    FIELD=PART_KEY,,A17,A17,$
    FIELD=FILLER1,,A27,A27,$
 $
    FIELD=RECTYPE,'PARTROOT', USAGE=A8, ACTUAL=A8,$
 $
    FIELD=PARTKEY,ALIAS=PARTKEY,A17,A17,INDEX=I,$
    FIELD=FIELD1,,A33,A33,$
 SEGMENT=STANINFO,PARENT=PARTROOT,SEGTYPE=S0,$
GROUP =GROUP2,ALIAS=KEY,A19,A19,$
    FIELD=PLACE_HOLDER,,A17,A17,$
    FIELD=STAN_KEY,,A02,A02,$
    FIELD=FILLER2,,A25,A25,$
$
    FIELD=RECTYPE,'STANINFO', USAGE=A8, ACTUAL=A8
$
    FIELD=STANKEY,ALIAS=STANKEY,A2,A2, INDEX=I ,$
    FIELD=FIELD2,,A3,A3,$
    FIELD=FIELD3,,A10,A10,$
    FIELD=FIELD4,,A70,A70,$
 SEGMENT=STOKSTAT,PARENT=PARTROOT,SEGTYPE=S0,$
  GROUP =GROUP3,ALIAS=KEY,A33,A33,$
     FIELD=PLACE_HOLDER,,A17,A17,$
    FIELD=STOCK_KEY,,A16,A16,$
    FIELD=FILLER3,,A11,A11,$
$
      FIELD=RECTYPE,'STOKSTAT', USAGE = A8, ACTUAL=A8,$
 $
      FIELD=STOCKEY,ALIAS=STOCKEY,A16,A16, INDEX=I ,$
      FIELD=FIELD5,,A10,A10,$
      FIELD=FIELD6,,A114,A114,$
 SEGMENT=CYCCOUNT,PARENT=STOKSTAT,SEGTYPE=S0,$
    GROUP =GROUP5,ALIAS=KEY,A35,A35,$
    FIELD=PLACE_HOLDER,,A33,A33,$
    FIELD=CYC_KEY,,A02,A02,$
    FIELD=FILLER7,,A9,A9,$
 $
    FIELD=RECTYPE,'CYCCOUNT',USAGE = A8,ACTUAL=A8,$
 $
    FIELD=CYCKEY,ALIAS=CYCCKEY,A2,A2, INDEX=I ,$
    FIELD=FIELD7,,A10,A10,$
    FIELD=FIELD8,,A10,A10,$
```

```
      FIELD=FIELD9,,A13,A13,$
 SEGMENT=BACKORDR,PARENT=STOKSTAT,SEGTYPE=S0,$
    GROUP =GROUP6,ALIAS=KEY,A44,A44,$
    FIELD=PLACE_HOLDER,,A33,A33,$
    FIELD=BACK_KEY,,A11,A11,$
$
   FIELD=RECTYPE,'BACKORDR', USAGE = A8, ACTUAL=A8,$
$
    FIELD=BACKEY,BACKEY,A11,A11, INDEX=I ,$
    FIELD=FIELD10,,A65,A65,$
```

TPDATA Access Files

The TPDATA Access File contains several parameters that identify the following for the Client Interface:

- The TPDATA server program to execute in CICS.
- The communication method.
- A transaction governor.
- User ID and password for Mainframe FOCUS

A value of TPVSAM for the parameter PARTNER_PROG specifies access to CICS/VSAM data. A value of TPIMSDB,$ for the parameter PARTNER_PROG specifies access to CICS/IMSDB data files.

Syntax Access File for TPVSAM and TPIMSDB

```
PARTNER_PROG=prog,$
COMM_METHOD=protocol,$

PARTNER_NAME=hostname,$
PARTNER_PORT=port,$

EDA_PROFILE=node_name

PARTNER_LU=partlu
LOGMODE=lmode

PREALLOC=prealloc,$
TRANLIM=tranlim,$


USER_ID=userid,$

PASSWORD=pw
```

where:

`prog`

Identifies the data to be accessed. This parameter is required. Valid values are:

```
TPVSAM              Specifies access to CICS/VSAM data files.
TPIMSDB             Specifies access to CICS/IMS data files.
```

`protocol`

Identifies the protocol used to communicate with the CICS region. Valid values are `TCPIP` or `EDAAPI`. For Mainframe FOCUS, TCPIP must be specified. For communicating with an EDA Server on MVS, UNIX, or NT, EDAAPI is used to reference the communication information that is entered in the ODIN.CFG (communications configuration file). This parameter is required.

`hostname`

Indicates the MVS or VSE host name where the CICS region resides. Required when COMM_METHOD=TCPIP. This parameter is required in the Access File for Mainframe FOCUS.

`port`

Indicates the port number defined to the EDA/TP Gateway in the CICS region. Required when COMM_METHOD=TCPIP. This parameter is required in the Access File for Mainframe FOCUS.

`node_name`

Identifies the client node name that controls inbound or outbound communications. Required when COMM_METHOD=EDAAPI. Node_name corresponds to the node_name in the ODIN.CFG configuration file.

`partlu`

Indicates the logical unit with which to communicate. Required when COMM_METHOD=EDAAPI and APPC is the communications method. This unit ultimately communicates with the CICS region.

`lmode`

Identifies the log LU log mode. Required when COMM_METHOD=EDAAPI *and* APPC is the communications method.

`prealloc`

Identifies the size of the buffer in bytes that is sent across the wire for an individual request. This parameter is used to set up storage in the CICS region for each transaction and storage in the TPDATA client, thus affecting the memory utilization of each CICS transaction. The size of this buffer limits the number of records that are passed from CICS to the client. If this parameter is used when COMM_METHOD=TCPIP, the number of records retrieved is approximately the result of the number of bytes per buffer (PREALLOC) divided by the average record length of the VSAM record. If this parameter is used when COMM_METHOD=EDAAPI, the buffer size specified in the Access File overrides the specification in the ODIN.CFG file. Required when COMM_METHOD = TCPIP. The default is 40K. This parameter at the minimum should at least be the size of the largest VSAM record that will be accessed. The number of bytes must be equal or greater than 4096. A specification of less than 4096 will return an error.

`tranlim`

Indicates the maximum number of CICS transactions to initiate in the CICS region to fulfill the request. Use this parameter to limit the number CICS transactions executed for a request. Each CICS transaction attempts to retrieve the number of records that can fit within the buffer specified in PREALLOC. The default value is 10 transactions. TRANLIM is used as the parameter to control the TPDATA Transaction Governor. See *TPDATA Transaction Governor* for additional informatio***n.***

`userid`

Indicates the User ID that connects to the CICS region. If CICS security is set up, this user must be authorized to logon to the CICS region and access the VSAM file.

When COMM_METHOD=EDAAPI, this parameter is specified in the EDASPROF.PRF file using the SQL EDA SET USER command. Required for Mainframe FOCUS.

`pw`

Indicates the password for the associated User ID that connects to the CICS region. If CICS security is set up, this user must be authorized to logon to the CICS region and access the VSAM file. When COMM_METHOD=EDAAPI, this parameter is specified in the EDASPROF.PRF file using the SQL EDA SET USER command. Required for Mainframe FOCUS clients.

Note: When COMM_METHOD=EDAAPI, see *Configuring Communications With an EDA Server* for additional informatio***n.***

Example Sample TPDATA VSAM Access File

The following is a sample Mainframe FOCUS Access File for the CICS/VSAM file FILEA.

When COMM_METHOD=EDAAPI, some of these parameters are specified in the ODIN.CFG configuration file. See *Configuring Communications With an EDA Server.* For a sample VSAM Access File member, use TPFILFCX which was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

The following Access File member TPFILACX was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

```
****************************************
* Example TPVSAM Access file for VSAM file FILEA using
 * native TCPIP access (Mainframe FOCUS)
 ****************************************
 PARTNER_PROG=TPVSAM,$
COMM_METHOD=TCPIP,$
 PARTNER_NAME=hostname,$
 PARTNER_PORT=port,$
 PREALLOC = 45050,$
* Transaction limit  optional.  Default = 10 transactions
*TRANLIM=nn,$
USER_ID=EDATP,$
PASSWORD=TPPSWD,$
```

Example Sample TPDATA IMS Access File

The Access File member TPD21ACX was unloaded as part of the EDA/TP Gateway for CICS installation procedure. The partner_prog is TPIMSDB and includes two other parms:

```
PSBNAME=psbname,$
PCBNUMBER=pcbnum,$
```

where:

`psbname`

Identifies the IMS PSB for the file. Required.

`pcbnum`

Identifies the position of the PCB within the PSB including I/O PCBs. Required.

When COMM_METHOD=EDAAPI, some of the Access File parameters are specified in the ODIN.CFG configuration file. See *Configuring Communications With an EDA Server.* For a sample IMS Access File, use member TPD21FCX which was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

Configuring Communications With an EDA Server

If you are running a TPDATA Interface located on an EDA Server you need to establish communication with the EDA Server. This is accomplished by adding the following code to the TPDATA Access File located on the server:

`COMM_METHOD=EDAAPI`

This code entry in the Access File instucts the program to refer to the ODIN.CFG file for further information about configuring the communication.

The TPDATA EDAAPI communication method supports three communication protocols: TCPIP, APPC (LU6.2), and MQSERIES. This document provides details for TCPIP. To configure Windows NT client for APPC or MQSERIES, see the *EDA/TP Gateway User Guide.*

Note: The EDA Server that is used to process the TPDATA requests must be installed on the desired platform. For details see the EDA Server installation documentation that applies to your platform.

Windows NT EDA Server configuration requirements are equivalent for the following IBI application tools, FOCUS Desktop, Cactus, Cactus Servers on UNIX or MVS, and for other EDA client applications.

The server configuration can be performed by executing the appropriate setup program and by manually modifying the EDASPROF.PRF and ODIN.CFG files. For details see *Modifying the EDASPROF. PRF and ODIN.CFG Files*.

Modifying the EDASPROF. PRF and ODIN.CFG Files

When an EDA Server is installed, the EDASPROF. PRF and ODIN.CFG files are created automatically and located in the following directories:

- For FOCUS Desktop

    `x:\IBI\FOCUS7\CONF\ETC\ directory`

- For an EDA Server

    `x:\IBI\EDASERVER42\CONF\ETC\ directory r`

You must then modify the files to include specific TPDATA Interface information.

For EDASPROF.PRF, add the following syntax as the final line in the file:

```
SQL EDA SET USER node_name/id,password
```

This entry provides the userid and password that are requred to sign-on to the CICS region:

For ODIN.CFG, add the following syntax for TYPE and PREALLOC after the SERVICE setting:

```
NODE= node_name
BEGIN
  PROTOCOL              = protocol
  HOST= host name
  SERVICE= port
  TYPE                  = TP
  PREALLOC              = 45050
  CLASS            = CLIENT
END
```

These lines are required to access TPDATA.

where:

node_name

> Identifies the client node name that controls inbound or outbound communications. Corresponds to the node_name in the TPDATA Access File, which is displayed in the SERVER field.

protocol

> Identifies the protocol used to communicate with the CICS region. For values other than TCP, such as APPC or MQSERIES, refer to the appropriate EDA Server installation and configuration documentation and the *EDA/TP Gateway for CICS Users Manual* for software compatibility.

hostname

> Indicates the MVS or VSE host name where the CICS region resides. This parameter is required in the Access File for Mainframe FOCUS.

port

> Indicates the TCPIP port number defined to the EDA/TP Gateway in the CICS region. This parameter is required in the Access File for Mainframe FOCUS.

PREALLOC

> Contains a description found in the TPVSAM or TPIMS Access file, such as 45050 in this illustration.

Example **Sample TPDATA VSAM Master File**

The following is a sample Master File for the CICS/VSAM file FILEA. The Master File member TPFILMAS was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

The CODEPAGE value of 37 is required to allow for data conversion. Suffix=TPVSAM executes the TPVSAM Interface.

This Master File must be transfered to the appropriate server platform or manually created in the proper client environment. For an EDA MVS server, the Master and Access Files are allocated to the libraries allocated to EDAMFD and EDAAFD respectively. For a non-MVS server, an allowable name might be FILEA.MAS. Note that the corresponding Access File is named FILEA.ACX.

```
FILE=FILEA,SUFFIX=TPVSAM,CODEPAGE=37,$
$FILE=FILEA,SUFFIX=VSAM,$
SEGNAME=FILEA,SEGTYPE=S0,$
 FIELD=FILL, ,A1,  A1,$
 GROUP=IDKEY,ALIAS=KEY,USAGE=A6,  ACTUAL=A6,$
 FIELD=ID1, ID1A,A3,  A3,$
 FIELD=ID2, ID2A,A3,  A3,$
 FIELD=NAME, NAME, A20, A20,$
```

```
FIELD=ADDR, ADDR, A20, A20,$
FIELD=PHONE, PHN, A10, A10,$
FIELD=X, X1, A1, A1,$
FIELD=ALTNDX, FILEAALT, A2, A2,INDEX=I,$
FIELD=Y, Y1, A1, A1,$
FIELD=NUM1, NUM1, A2, A2,$
FIELD=AMT, AMT, A8, A8,$
FIELD=SPLAT,  STAR, A9,  A9,$
```

Example **Sample TPDATA VSAM Access File**

The following is a sample Access File for the CICS/VSAM file FILEA. The Access File member TPFILFCX was unloaded as part of the EDA/TP Gateway for CICS installation procedure.

This Access File must be transfered to the appropriate server platform or manually created in the proper client environment. For an EDA MVS Server, the Master and Access Files are allocated to the libraries allocated to EDAMFD and EDAAFD respectively. For non-MVS server, an allowable name might be FILEA.ACX. Note that the corresponding Master File is named FILEA.MAS.

```
PARTNER_PROG=TPVSAM,$
COMM_METHOD=EDAAPI,$
SERVER=TPSERV,$
* the following parameters are for APPC
*PARTNER_LU=partlu
*LOGMODE=lmode
* Must equal PREALLOC parm in ODIN.CFG file
PREALLOC=45050,$
* Transaction limit optional. default = 10 transactions
*TRANLIM=nn,$
```

Example **Sample ODIN.CFG Fle**

The following is an example of a communications configuration file for an EDA Server. All the following parameters are required to successfully establish communication with the CICS region on the mainframe.

```
; ODIN config file generated on 12/18/1998 16.37.40
NODE= TPSERV
BEGIN
  PROTOCOL            = TCP
  HOST                = IBIMVSM
  SERVICE        = 1111
  TYPE                = TP
  PREALLOC            = 45050
  CLASS          = CLIENT
END
```

Note: NODE in the ODIN.CFG configuration file must match SERVER in the Access File. In these examples both entries are TPSERV.

Example **Sample EDASPROF.PRF File for Windows NT Clients**

The following is an example of a communications configuration file for an EDA Server. All the following parameters are required to successfully connect to the CICS region on the mainframe.

```
-*********************************************************
-* Profile generated on 18 December 1998 at 16:37:40
-*********************************************************
-*
-* SET EDAPATH=                                  -
-*             <path1>;                          -
-*             <path2>;                          -
-*               ... ;                           -
-*             <pathN>
-* SET EXORDER=settings
-*
SQL EDA SET JOINTYPE SORTMERGE
SQL EDA SET USER TPSERV/id,password
-*
```

Note: The TPSERV entry in the added line of code corresponds to the node_name entered in the ODIN.CFG file and the SERVER entry in the Access File.

TPDATA Transaction Governor

The TPDATA Interface is a transaction-based Interface designed primarily for high volume of small transactions. Keyed and index access to the VSAM, IMS or DB2 files help to ensure that the records processed by CICS will be limited to a small answer set. The more limited the answer set is, the less impact on the rest of the CICS region. A small answer set can be achieved by the use of IF or WHERE tests on appropriate keys and indexes.

The TPDATA Interface issues calls to the EDA/TP Gateway transaction for all retrieval and update requests. Each call to the EDA/TP Gateway results in a CICS transaction. The Interface will continue issuing CICS transactions until the retrieval or update request is done. Ad-hoc reporting and file joins can adversely affect the CICS or IMS/TM region because of the potential of large amounts of transactions. It is very important to monitor the affect of such reporting applications. See *Limiting the Affects of Joins*.

TRANLIM. The TPDATA Interface has the capability to limit the number of CICS or IMS transactions that are executed per request. The TRANLIM parameter in the Access File or ODIN configuration file is used to specify the maximum number of transactions to be executed for one complete MAINTAIN, SQL, TABLE or MODIFY request. The TRANLIM parameter specified in the Access File overrides the specification in the ODIN configuration file. If the number of transactions needed to fulfill the retrieval or update request is greater than the specified TRANLIM parameter, the request is terminated and the appropriate error message is displayed.

Limiting the Affects of JOINS

The TPDATA Interface does not prohibit the execution of JOINS. However, the user must exercise caution when joining files. A CICS or IMS transaction is executed for every cross-referenced (joined-to) record. The number of transactions executed for each record in the joined-to file will equal the number of records returned from the host (parent) file. If the host file answer set contains 1 million rows then 1 million CICS or IMS transactions will be needed to attempt to retrieve the joined-to records.

There are several implications that need to be noted.

1. The choice of the parent or host file is crucial. IF or WHERE tests on the host file will limit the returned answer set, thus limiting the transactions needed to retrieve the cross-referenced records. With this in mind, take a look at the order of the files in the JOIN, the switching of the order may allow the request to run faster with much less CICS or IMS overhead.

2. The access file or ODIN.CFG file TRANLIM parameter will limit the number of transactions needed to complete all requests including JOINS. The TRANLIM parameter can be used to limit the affects of the join transactions.

Index